



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ

«Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 14

Студент: Керимов А. Ш.

Группа: ИУ7-64Б

Преподаватель: Толпинская Н. Б.

Москва.
2020 г.

Цель работы — получить навыки построения модели предметной области, разработки и оформления программы на Prolog, изучить принципы, логику формирования программы и отдельные шаги выполнения программы на Prolog.

Задание. Используя базу знаний, хранящую знания (лаб. 13):

- **«Телефонный справочник»:** Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
- **«Автомобили»:** Фамилия_владельца, Марка, Цвет, Стоимость, и др.,
- **«Вкладчики банков»:** Фамилия, Банк, счет, сумма, др.

Владелец может иметь несколько телефонов, автомобилей, вкладов (Факты). В разных городах есть однофамильцы, в одном городе – фамилия уникальна.

Используя **конъюнктивное правило и простой вопрос**, обеспечить возможность поиска:

По Марке и Цвету автомобиля найти Фамилию, Город, Телефон и Банки, в которых владелец автомобиля имеет вклады. Лишней информации не находить и не передавать!!!

Владельцев может быть **несколько** (не более 3-х), **один** и **ни одного**.

1. Для каждого из трех вариантов **словесно подробно** описать порядок формирования ответа (в виде таблицы). При этом, указать – отметить моменты очередного запуска алгоритма унификации и полный результат его работы. Обосновать следующий шаг работы системы. Выписать унификаторы – подстановки. Указать моменты, причины и результат отката, если он есть.
2. Для случая нескольких владельцев (2-х):
приведите примеры (таблицы) работы системы **при разных порядках** следования в БЗ процедур, и знаний в них: («Телефонный справочник», «Автомобили», «Вкладчики банков», или: «Автомобили», «Вкладчики банков», «Телефонный справочник»). Сделайте **вывод**: Одинаковы ли: множество работ и объем работ в разных случаях?
3. Оформите 2 таблицы, демонстрирующие **порядок работы алгоритма унификации** вопроса и подходящего заголовка правила (для двух случаев из пункта 2) и укажите результаты его работы: ответ и побочный эффект.

Практическая часть

Листинг 1. Телефонный справочник, автомобили и вклады

```
domains
surname, phone, city, street, house, apartment = symbol
address = address(city, street, house, apartment)
model, color, cost = symbol
bank, account, sum = symbol
```

```

predicates
subscriber(surname, phone, address)
car(surname, model, color, cost)
deposit(surname, bank, account, sum)

find_surname_city_phone_bank_by_model_color(model, color, surname, city, phone, bank)

clauses
subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")).
subscriber("Yugay", "2222222222", address("Lyubertsy", "Yubileynaya", "17b", "18")).
subscriber("Novikov", "4444444444", address("Moscow", "Armyanskaya", "1", "1")).

car("Yugay", "Nissan", "Red", "1000000").
car("Kerimov", "Volvo", "Blue", "1200000").
car("Novikov", "Nissan", "Red", "1100000").

deposit("Kerimov", "Sberbank", "0000-0000", "1000000").
deposit("Novikov", "VTB", "0001-0000", "300000").
deposit("Yugay", "Sberbank", "0000-0001", "500000").

find_surname_city_phone_bank_by_model_color(Model, Color, Surname, City, Phone, Bank) :-
    car(Surname, Model, Color, _),
    subscriber(Surname, Phone, address(City, _, _, _)),
    deposit(Surname, Bank, _, _).

goal
find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank).
%find_surname_city_phone_bank_by_model_color("Volvo", "Blue", Surname, City, Phone, Bank).
%find_surname_city_phone_bank_by_model_color("Volvo", "Red", Surname, City, Phone, Bank).

```

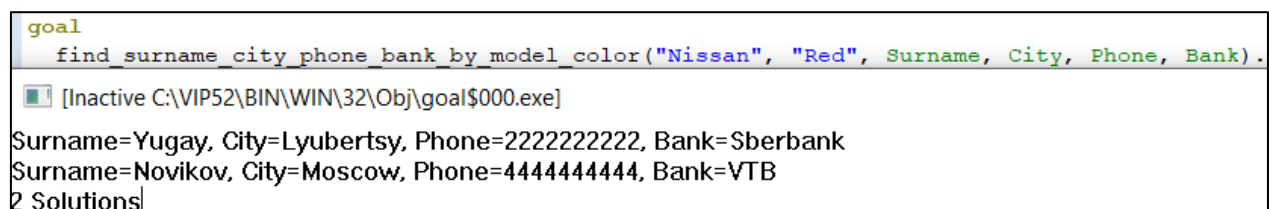


Рисунок 1 – Несколько владельцев

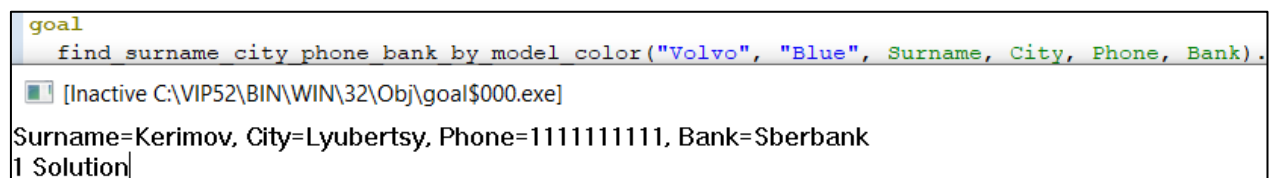


Рисунок 2 – Один владелец

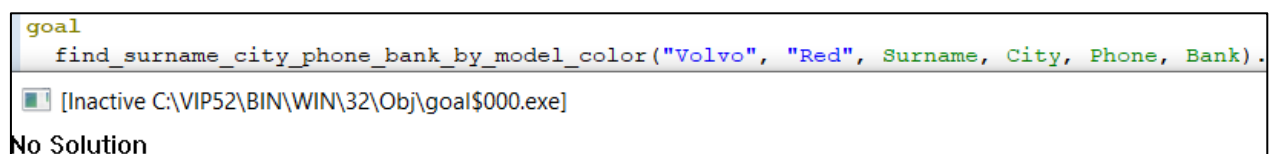


Рисунок 3 – Ни одного владельца

(Задание 1) Подробное описание порядка формирования ответа из рисунка 1.

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия
1	Сравнение find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) и subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")). Функторы не равны	Прямой ход, к следующему правилу
...

10	<p>Сравнение find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) и find_surname_city_phone_bank_by_model_color(Model, Color, Surname, City, Phone, Bank) :- car(Surname, Model, Color, _), subscriber(Surname, Phone, address(City, _, _, _)), deposit(Surname, Bank, _, _).</p> <p>Функторы равны, Model конкретизируется "Nissan", Color конкретизируется "Red".</p> <p>Сцепление Surname к Surname, City к City, Phone к Phone, Bank к Bank.</p>	<p>Прямой ход, к унификации car(Surname, "Nissan", "Red", _)</p>
11	<p>Сравнение car(Surname, "Nissan", "Red", _) и subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")).</p> <p>Функторы не равны</p>	<p>Прямой ход, к следующему правилу</p>
...
14	<p>Сравнение car(Surname, "Nissan", "Red", _) и car("Yugay", "Nissan", "Red", "1000000"). Функторы равны, термы "Nissan" равны, термы "Red" равны, Surname конкретизируется "Yugay".</p>	<p>Прямой ход к унификации subscriber(Surname, Phone, address(City, _, _, _)).</p> <p>Surname = "Yugay"</p>
15	<p>Сравнение subscriber("Yugay", Phone, address(City, _, _, _)) и subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")).</p> <p>Функторы равны, термы "Yugay" и "Kerimov" не равны.</p>	<p>Прямой ход, к следующему правилу Surname = "Yugay"</p>
16	<p>Сравнение subscriber("Yugay", Phone, address(City, _, _, _)) и subscriber("Yugay", "2222222222", address("Lyubertsy", "Yubileynaya", "17b", "18")).</p> <p>Функторы равны, термы "Yugay" равны, Phone конкретизируется "2222222222", City конкретизируется "Lyubertsy".</p>	<p>Прямой ход, к унификации deposit("Yugay", Bank, _, _).</p> <p>Surname = "Yugay", Phone = "2222222222", City = "Lyubertsy".</p>
17	<p>Сравнение deposit("Yugay", Bank, _, _) и subscriber("Kerimov", "1111111111", address("Lyubertsy", "Yubileynaya", "17", "2")).</p> <p>Функторы не равны</p>	<p>Прямой ход, к следующему правилу. Surname = "Yugay", Phone = "2222222222", City = "Lyubertsy".</p>
...
25	<p>Сравнение deposit("Yugay", Bank, _, _) и deposit("Yugay", "Sberbank", "0000-0001", "500000").</p> <p>Функторы равны, термы "Yugay" равны, Bank конкретизируется "Sberbank".</p>	<p>Откат, реконкретизация Bank и переход к следующему правилу. 1 решение найдено Surname = "Yugay", Phone = "2222222222", City = "Lyubertsy", Bank = "Sberbank"</p>

26	Сравнение deposit("Yugay", Bank, _, _) и find_surname_city_phone_bank_by_model_color(Model, Color, Surname, City, Phone, Bank) :- car(Surname, Model, Color, _), subscriber(Surname, Phone, address(City, _, _, _)), deposit(Surname, Bank, _, _). Функторы не равны	Откат, реконкретизация Phone и City, переход к следующему правилу (относительно 16). Surname = "Yugay", Phone = "2222222222", City = "Lyubertsy".
27	Сравнение subscriber("Yugay", Phone, address(City, _, _, _)) и subscriber("Novikov", "4444444444", address("Moscow", "Armyanskaya", "1", "1")). Функторы равны, термы "Yugay" и "Novikov" не равны.	Прямой ход, к следующему правилу. Surname = "Yugay".
...
34	Сравнение subscriber("Yugay", Phone, address(City, _, _, _)) и find_surname_city_phone_bank_by_model_color(Model, Color, Surname, City, Phone, Bank) :- car(Surname, Model, Color, _), subscriber(Surname, Phone, address(City, _, _, _)), deposit(Surname, Bank, _, _). Функторы не равны	Откат, реконкретизация Surname и переход к следующему предложению (относительно 14). Surname = "Yugay".
35	Сравнение car(Surname, "Nissan", "Red", _) и car("Kerimov", "Volvo", "Blue", "1200000"). Функторы равны, термы "Nissan" и "Volvo" не равны.	Прямой ход, к следующему предложению.
36	Сравнение car(Surname, "Nissan", "Red", _) и car("Novikov", "Nissan", "Red", "1100000"). Функторы равны, термы "Nissan" равны, термы "Red" равны, Surname конкретизируется "Novikov".	Прямой ход, к унификации subscriber("Novikov", Phone, address(City, _, _, _)). Surname = "Novikov".
...	... Далее аналогично программа найдёт второе решение Surname="Novikov", City="Moscow", Phone="4444444444", Bank= "VTB", попытается найти третье, не найдёт, выведет два решения и завершится.	...

(Задание 2) Множество работ и объём работ в разных случаях (при разных порядках следования в БЗ процедур и знаний в них) одинаковы. Это очевидно, так как на каждом этапе доказательства происходит полный перебор, при котором порядок не важен.

(Задание 3)

Таблица 1

Ш	Результирующая ячейка	Рабочее поле	П	Стек
а			у	
г			н	
у			к	
н			т	
и			а	
ф			л	
и			г	
к			о	
а			р	
ц			и	

и и			Т М а	
1			1	find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) = find_surname_city_phone_bank_by_model_color("Nissan", "Red", "Yogay", "Lyubertsy", "222222222", "Sberbank")
2		find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) = find_surname_city_phone_bank_by_model_color("Nissan", "Red", "Yogay", "Lyubertsy", "222222222", "Sberbank")	е	"Nissan" = "Nissan" "Red" = "Red" Surname = "Yugay" City = "Lyubertsy" Phone = "222222222" Bank = "Sberbank"
3	"Nissan" = "Nissan"	"Nissan" = "Nissan"	б	"Red" = "Red" Surname = "Yugay" City = "Lyubertsy" Phone = "222222222" Bank = "Sberbank"
4	"Nissan" = "Nissan" "Red" = "Red"	"Red" = "Red"	б	Surname = "Yugay" City = "Lyubertsy" Phone = "222222222" Bank = "Sberbank"
5	"Nissan" = "Nissan" "Red" = "Red" Surname = "Yugay"	Surname = "Yugay"	г	City = "Lyubertsy" Phone = "222222222" Bank = "Sberbank"
6	"Nissan" = "Nissan" "Red" = "Red" Surname = "Yugay" City = "Lyubertsy"	City = "Lyubertsy"	г	Phone = "222222222" Bank = "Sberbank"
7	"Nissan" = "Nissan" "Red" = "Red" Surname = "Yugay" City = "Lyubertsy" Phone = "222222222"	Phone = "222222222"	г	Bank = "Sberbank"
8	"Nissan" = "Nissan" "Red" = "Red" Surname = "Yugay" City = "Lyubertsy" Phone = "222222222" Bank = "Sberbank"	Bank = "Sberbank"	г	
В ы в о д	подстановка	Т.к. стек пуст — успех и в результирующей ячейке подстановка		

Таблица 2

Шаг унификации	Результатирующая ячейка	Рабочее поле	Пункт алгоритма	Стек
1			1	find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) = subscriber(surname, phone, address (city, street, house, apartment))
2		find_surname_city_phone_bank_by_model_color("Nissan", "Red", Surname, City, Phone, Bank) = subscriber(surname, phone, address (city, street, house, apartment))	д	
Вывод		Неудача, унификация невозможна		

Теоретическая часть

В какой части правила сформулировано знание? Это знание о чём, с формальной точки зрения?

Знание содержится в заголовке правила. С формальной точки зрения, это знание о том, что в заголовке, являющимся составным термом $f(t_1, \dots, t_m)$, между аргументами t_1, \dots, t_m существует отношение, а имя этого отношения — f .

Что такое процедура?

Процедурой в Prolog называется совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же аргументность. Процедура описывает совокупность правил отношения.

Сколько в БЗ текущего задания процедур?

4.

Что такое пример терма, это частный случай терма, пример? Как строится пример?

Пусть $\Theta = \{x_1 = t_1, x_2 = t_2, \dots, x_n = t_n\}$ — подстановка, тогда результат применения подстановки к терму обозначается: $A\Theta$. Применение подстановки заключается в замене каждого вхождения переменной x_i на соответствующий терм. Терм B называется примером терма A , если существует такая подстановка Θ , что $B = A\Theta$.

Что такое наиболее общий пример?

Терм C — наиболее общий пример термов A и B , если существуют такие подстановки Θ_1 и Θ_2 , что $C = A\Theta_1$ и $C = B\Theta_2$.

Назначение и результат работы алгоритма унификации.

Унификация — операция, которая позволяет формализовать процесс логического вывода (наряду с правилом резолюции). С практической точки зрения — это основной вычислительный шаг, с помощью которого происходит: двунаправленная передача параметров процедурам, неразрушающее присваивание, проверка условий (доказательство).

В каком случае запускается механизм отката?

Механизм отката запускается в случае, если унификация завершилась неудачей.

Виды и назначение переменных в Prolog. Примеры из задания. Почему использованы те или другие переменные (примеры из задания)?

Назначение переменных — передача информации «во времени и в пространстве». Виды переменных: именованные и анонимные. Именованная переменная уникальна в пределах предложения, анонимная — любая уникальна. В процессе выполнения программы именованные переменные могут конкретизироваться (связываться со значением). Неименованные переменные не связываются со значениями. Переменные в ходе выполнения программы могут реконкретизироваться.

```
find_surname_city_phone_bank_by_model_color(Model, Color, Surname, City, Phone, Bank) :-
    car(Surname, Model, Color, _),
    subscriber(Surname, Phone, address(City, _, _)),
    deposit(Surname, Bank, _, _).
```

Здесь анонимные переменные используются, так как не важно, какие именно объекты им соответствуют при доказательстве (не находится и не передаётся лишняя информация).